DOCUMENT RESUME

ED 459 818 IR 058 354

AUTHOR Francisco-Revilla, Luis; Shipman, Frank; Furuta, Richard; Karadkar, Unmil; Arora, Avital

TITLE Managing Change on the Web.
SPONS AGENCY National Science Foundation, Arlington, VA.
PUB DATE 2001-06-00
NOTE 11p.; In: Proceedings of the ACM/IEEE-CS Joint Conference on Digital Libraries (1st, Roanoke, Virginia, June 24-28, 2001). For entire proceedings, see IR 058 348. Figures may not reproduce well.
CONTRACT IIS-9812040; DUE-0085798
AVAILABLE FROM Association for Computing Machinery, 1515 Broadway, New York NY 10036. Tel: 800-342-6626 (Toll Free); Tel: 212-626-0500; e-mail: acmhelp@acm.org. For full text: http://www1.acm.org/pubs/contents/proceedings/dl/379437/.
PUB TYPE Reports - Research (143) -- Speeches/Meeting Papers (150)
EDRS PRICE MF01/PC01 Plus Postage.
DESCRIPTORS *Change; Computer System Design; Electronic Libraries; *Information Management; Information Retrieval; Internet; Library Collection Development; Man Machine Systems; Relevance (Information Retrieval); User Satisfaction (Information); *World Wide Web
IDENTIFIERS *Web Page Design; *Web Pages

ABSTRACT

Increasingly, digital libraries are being defined as collection pointers to World Wide Web-based resources rather than collections that hold the resources themselves. Maintaining these collections is challenging due to distributed document ownership and high fluidity. Typically a collection's maintainer has to assess the relevance of changes with little system aid. This paper describes the Walden's Paths Path Manager, which assists a maintainer in discovering when relevant changes occur to linked resources. The approach and system design was informed by a study of how people perceive changes of Web pages. The study indicated that structural changes are key in determining the overall change and that presentation changes are considered irrelevant. The paper is divided into seven sections. The first section is an introduction. The second presents background about this work and its motivation. Sections three and four describe the approach used and its implementation as a prototype system. Section five gives an overview of the study to better understand how people perceive changes and how to refine the Path Manager. Sections six and seven conclude the paper. (Contains 24 references.) (AEF)

Reproductions supplied by EDRS are the best that can be made
from the original document.

Managing Change on the Web

Luis Francisco-Revilla, Frank Shipman, Richard Furuta, Unmil Karadkar, and Avital Arora

Center for the Study of Digital Libraries and Department of Computer Science
Texas A&M University
College Station, TX 77843-3112, USA

{l0f0954, shipman, furuta, unmil, avital}@csdl.tamu.edu

PERMISSION TO REPRODUCE AND DISSEMINATE THIS MATERIAL HAS BEEN GRANTED BY

D. Cotton

TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)

U.S. DEPARTMENT OF EDUCATION
Office of Educational Research and Improvement
EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)

☐ This document has been reproduced as received from the person or organization originating it.

☐ Minor changes have been made to improve reproduction quality.

• Points of view or opinions stated in this document do not necessarily represent official OERI position or policy.

1

ABSTRACT

Increasingly, digital libraries are being defined that collect pointers to World-Wide Web based resources rather than hold the resources themselves. Maintaining these collections is challenging due to distributed document ownership and high fluidity. Typically a collection's maintainer has to assess the relevance of changes with little system aid. In this paper, we describe the Walden's Paths Path Manager, which assists a maintainer in discovering when *relevant* changes occur to linked resources. The approach and system design was informed by a study of how humans perceive changes of Web pages. The study indicated that structural changes are key in determining the overall change and that presentation changes are considered irrelevant.

Categories and Subject Descriptors

I.3.7 **[Digital Libraries]**: User issues;
H.5.4 **[Hypertext/Hypermedia]**: Other (maintenance)

General Terms

Algorithms, Management, Design, Reliability, Experimentation, Human Factors, Verification.

Keywords

Walden's Paths, Path Maintenance.

1. INTRODUCTION

The work in building a library is not only in the collection of materials, it is also in their organization and maintenance.

Books in a traditional library are actively managed—when acquired they must be organized, indexed and catalogued. When considered obsolete, they must be removed. Librarians go through great length in order to keep the collection up to date. New editions replace old ones, while old versions may be moved to archival sections or be discarded.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
JCDL'01, June 24-28, 2001, Roanoke, Virginia, USA.
Copyright 2001 ACM 1-58113-345-6/01/0006...$5.00.

Digital libraries present an electronic counterpart of traditional libraries. Digital librarians are necessary to perform similar functions in order to maintain collections of electronic documents or electronic pointers to documents.

It is not rare to hear people refer to the Web as a new kind of library—or at least that it should be more like a library. However on the Web, not all electronic collections are well maintained—the degree of management of electronic collections varies considerably. Moreover, the characteristics of the Web raise many challenges for the maintainer. Collections on the Web can be extremely distributed—not only the location of the documents is distributed, but ownership is also distributed.

At the moment, the Web is less like a library and more like a huge bookshelf, containing millions of documents and links to documents. This mega shelf is subject to the activities of an army of people (and computer-based agents) that keep acting upon it in ways such as adding, removing, indexing, cataloguing, rearranging, modifying, and copying documents. Additionally, all these activities tend to occur with little or no coordination, worsening the situation.

Better coordination, social protocols, and institutionalized management of collection can alleviate the situation in some cases, particularly when the owner of the collection and the owner(s) of the documents agree to coordinate their activities. The emerging digital topical libraries, while homed on the Web, are better maintained than the Web as a whole. They present a more cohesive and organized structure in order to support their patrons. Different automatic mechanisms have been devised in order to help manage the documents. Indexing and cataloguing are considerably easier to perform than in the pre-electronic days. Nevertheless, considerable human effort is still required to monitor the changes in documents and collections.

In addition to the digital library that is provided by the "owner" of the documents (e.g., the ACM Digital Library) there are a growing number of specialized libraries with distributed ownership.[1] In these cases librarians often are limited to working with the pointers or links to the documents.

Digital documents are typically very fluid, i.e., changes are a common occurrence. Although detecting changes is easily

[1] One example is the National Science Foundation's NSDL effort, described at http://www.ehr.nsf.gov/due/programs/nsdl/

BEST COPY AVAILABLE

ED 459 818

IR058354

automated, assessing the relevance of changes is not. Typically humans are needed when facing the task of relevance assessment. In fact, "relevance" is a highly abstract, contextual, and subjective concept. Different individuals can disagree in their relevance judgment about any given change. Nevertheless, as difficult as it may be, assessing the relevance of a change is still necessary.

This paper presents an approach to aid humans in managing fluid collections of documents, in particular Web pages that have been authored and are owned by third parties. The approach provides a way of automatically assessing the relevance of changes in the Web pages. This method is explored in the Path Manager, a system designed to infer relevance of changes to Web pages and to communicate this information to the collection's maintainer.

The paper is divided into seven sections. Section two presents background about this work and its motivation. Sections three and four describe the approach used and its implementation as a prototype system. Section five gives an overview of a study to better understand how humans perceive changes, which we conducted to better understand how to refine the Path Manager. Sections six and seven conclude the paper.

2. MOTIVATION

Web readers access a rich and vast collection of information resources authored and published by a multitude of sources. In this collection it is possible to find information on virtually every topic, varying in aspects such as point of view, validity, semantics, rhetoric, and contextual situation. In order to better exploit this collection, different systems have been developed with the goal of aiding readers by providing an interpretation or contextualization of Web pages.

Walden's Paths is an application that allows teachers to construct trails [2] or paths using Web pages authored by others [7], [19]. The paths represent a higher order construct, or meta-document, that organizes and adds contextual information to pages authored by others. However, meta-documents that are to have a lasting value need to adapt to changes in their components. Thus, paths must address the issue of the unpredictable changing of their fundamental building blocks (Web pages). This is particularly important given the high fluidity of Web pages [3], where authors are prone to change their pages just to maintain a "fresh" look and feel.

Early in the Walden's Paths project issues related to the fluidity of Web pages rapidly arose. We observed that it is a common occurrence for paths to include Web pages that have moved, changed or are no longer available [18]. As a result the collection of paths needed to be constantly revised and updated.

To date in the project, we have implemented a number of approaches that can help to reduce the effects of fluidity. A simple approach is to cache the pages. This approach is one way to address the fluidity issue and also can expedite delivery of the information, particularly in schools that have slow Internet connections. However, it became evident to us that a single caching approach could not solve the problem, since not all Web page changes are undesirable (some paths include pages that by nature change, such as the current weather or a newspaper Web page). Our approach, implemented through the Walden's Paths Authoring Tool, allows authors to specify the caching strategy for individual pages [12]. We note, however, that issues about versioning and intellectual property rights remain, especially when caching Web pages created by third parties.

A second approach might be to increase the fluidity of the paths themselves—perhaps showing only pages that have not moved or changed. Walden's Paths provides an implementation mechanism for this through ephemeral paths [6]. Ephemeral paths are paths that exist for only a short period of time as the result of some computation. Unfortunately the effectiveness of the technique is limited here since many path authors design their paths to have a rhetorical coherence based on the linearity imposed by the path mechanism. Hence, when some pages of the path are removed the rhetorical structure can be broken, rendering the path ineffective.

More importantly, caching and ephemeral paths only provide *mechanisms* for implementing ways of reacting to change. Humans must determine when significant change has occurred. Making this determination requires that the nature of the change be considered within the context of the maintainer's goals (i.e., that its *relevance* be considered). For instance, even though a page may change in appearance or wording, it might remain conceptually the same. For many applications, this would be an insignificant change.

Before proceeding, and in order to avoid possible confusion, it is important to define the different people and components involved in the problem. While the issues addressed in the present work relate to any meta-document application, our interest originated from research conducted in the Walden's Paths project. Therefore meta-documents are defined as "paths" even though this is not the only possible meta-document. Similarly, construction blocks are referred to as "Web pages" or "pages". The first group involved is the people that originally create and publish the Web pages. They are identified to as "page creators". Another group is the people that create the meta-documents or paths. This group is denoted in this document as "path authors" or just "authors". Additionally there is the group of people accessing or using the paths, which is designated as "readers". Finally there is the group of people who has to manage the collection of paths and are referred as "administrators". It is important to consider that any given person might act in any or multiple of these roles at any given time.

2.1 Assessing Change Relevance

We are pursuing approaches to assist path authors and administrators in managing continuously occurring changes in their selected collection of Web pages. While our emphasis is on paths, our approach will be equally applicable to other Web-based domains that need to assess the relevance of changes to Web pages authored by third parties. We would expect, for example, that simple modifications to the Path Manager would enable the maintenance of personal bookmark lists or other collections of resource links.

Simply detecting changes in a page is easy. The simplest implementation would track the "last modified" date returned by the HTTP server. A more accurate mechanism would retain cached copies of Web pages. It is easy to compare a document with a previously cached version and determine if there has been any change. In a similar way, it would be possible to check for changes within specific sections of the document. These cases would return a Boolean "yes/no" indication of change.

As mentioned before, a difficulty arises when attempting to determine the relevance of changes. The obvious approach would be to assess the magnitude or amount of change. Unfortunately, there is not an easily computed metric that provides a direct correlation between syntactic and semantic changes in a Web page

For instance, there is no clear relationship between the number of bytes changed and the relevance of the change to the reader. A large number of bytes changed might result from a page creator who restructures the spacing of a page's source encoding while maintaining the same content from a semantic and rhetorical point of view. Similarly a small number of bytes changed might result from the insertion of a few negations in the text that causes a complete reversal in meaning. On the other hand, we can think of other situations where a large number of byte changes correspond to the creation of a completely different page, and situations where a few bytes are changed when the page creator simply corrects minor spelling and grammatical errors.

The goal would be to efficiently obtain a measure of the semantic distance between two versions of a document. However, at the moment the best we could attempt is to infer the semantic distance based on the syntactic characteristics of the document. In order to accomplish this, some heuristic actions are reasonable. For example the document can be partitioned based on heuristics such as:

- Paragraphs tend to encapsulate concepts
- Different paragraphs tend to encapsulate different concepts

Another way to partition the document is to analyze the document encoding. Markup languages such as SGML and XML already specify structure rather than presentation.

Even though Web pages are encoded in HTML, analyzing the changes is not an easily automated task. While HTML provides some information about the structure of the document, it is commonly used for specifying presentation rather than structure. Adding to that, page creators use HTML in extremely different ways. Paragraphs, headings and other tags are used quite diversely. What conceptually constitutes a paragraph for one page creator might constitute several pages for another. This does not mean that HTML tags are useless for analyzing change. On the contrary, HTML tags and other features such as keywords can be used in order to infer the relevance of changes.

2.2 Related Work

Measuring the magnitude or relevance of changes in an automated fashion is not an easy task. Researchers have encountered this problem in a variety of contexts and their approaches have informed our work.

In his doctoral dissertation David Johnson created a system for authoring and delivering tutorials over the Web [10], [11]. Johnson designed a signature-based approach when he also faced the issue of ever-changing Web pages. Johnson's approach computed a "distance" between two versions of a document by employing weighted values for additions and deletions of paragraphs, headings and keywords. Based on this measurement, Johnson created a mechanism that notifies readers and even blocks Web pages from showing when the distance between the two versions reaches predetermined trigger levels. While his testing was satisfactory with a particular collection of Web pages there are a couple of issues that hinder exporting the approach to the World Wide Web. The first issue is the dependence of the distance measure on the arbitrary determination of the weights assigned to the addition and deletions. The use of these weights might not work for a different collection of Web pages. The second issue is the asymmetric nature of the distance measurement and its lack of normalization. That is, the distance from document A to document B might be different from the distance from document B to document A. In addition there are no normalized values for the distance. Therefore the distance between two documents could be a number like 0.5 or 20. This makes setting trigger levels an arbitrary matter.

There are other approaches that monitor features of Web pages at a fine-grained level. Currently it is possible to find Web-based systems that provide fine-grained monitoring of changes such as AIDE[4], URL-Minder [22] and WatzNew [23]. Systems like these allow monitoring text, links, images and keywords of a given Web page. However a typical critique is that cosmetic changes are reported to be as relevant as substantive content changes. In contrast, the Path Manager evaluates the relevance of the change with regard to the whole page.

The goal of the Path Manager refers to identifying "interesting" or relevant changes to Web pages. As such, its design has been informed by other research work dealing with identifying "interesting" Web pages. In particular, research in page relevance has helped point out possible features of a page that should be monitored more attentively.

Researchers at the University of California at Irvine have investigated the issues of identifying readers' interests and dealing with changing Web pages. They have developed systems such as Syskill and Webert [17] and the Do-I-Care-Agent (DICA) [20], [21]. Syskill and Webert is an agent designed to discover interesting pages on the Web. The approach in this system is to aid readers with long-term information seeking goals. DICA is an agent designed to monitor reader-specified Web pages and notify the reader of important changes to these pages. Both systems rely on user profiles intended to model their user's interests. By interacting with the readers, the agents learn more about the reader's interests. However Walden's Paths is designed for a different environment, more specifically an educational environment. In this case the paths are used as artifacts that provide guidance and direction to the readers. Thus, Web pages must maintain consistency with regard to the semantic and rhetoric composition of the path.

There are two other systems that, although slightly tangential to the present work, have influenced the design of the Path Manager. These systems are WebWatcher [1], [9] and Letizia [14]. Like URL-Minder and WatzNew, WebWatcher notifies the user whenever specified pages change. In addition, WebWatcher attempts to evaluate "how interesting" a given Web page would be for a given reader and provides navigation suggestions in real time, by annotating or adding links to the page. This approach explores the use of the knowledge embedded in the links and the text around the links in order to infer relevance. While Johnson rejected links for page distance analysis, the arguments forwarded by WebWatcher prompted the consideration of them as a metric in the Path Manager.

Letizia is an autonomous interface agent [15] that aids a reader in browsing the Web. Letizia uses a knowledge-based approach to respond to personal interests. Like WebWatcher, it also attempts to evaluate how interesting a given Web page would be for a given reader. While the reader is reading a Web page in Netscape, Letizia traverses the links in the page, retrieves the pages and analyzes them. Then it ranks the links based on how interesting the destination Web pages might be for the reader. Letizia's inference takes place on the client as opposed to WebWatcher where the process is conducted on the server.

3. APPROACH

As described, determining the relevance of changes to a document can only be done in the context of how that document is to be used. As this knowledge will not be available to the system supporting relevance assessment, our goal is to provide a variety of information about changes in a relatively concise interface. To do so, we use a variety of document signatures to recognize different types of change.

3.1 Kinds of Change

In order to infer change relevance, it is important to classify the nature of the change. There are four categories of change that we distinguish between: content or semantic, presentation, structural, and behavioral.

Content changes refer to modifications of the page contents from the reader's point of view. For example, a page created for a soccer tournament might be continuously updated as the tournament progresses. After the tournament has ended, the page might change to a presentation about the tournament results and sports injuries.

Presentation changes are changes related to the document representation that do not reflect changes in the topic presented in the document. For instance, changes to HTML tags can modify the appearance of a Web page while it otherwise remains the same.

Structural changes refer to the underlying connection of the document to other documents. As an example, consider changes in the link destinations of a "Weekly Hot Links" Web page. While this page might be conceptually the same, the fact that the destination of the links have changed might be relevant, even if the text of the links has not. Structural changes are also important to detect, as they often might not be visually perceptible.

Behavioral changes refer to modifications to the active components of a document. For Web pages this includes scripts, plug-ins and applets. The consequences of these changes are harder to predict, especially since many pages hide the script code in other files.

3.2 Document Signatures

To represent the different characteristics of a Web-based document, we use a set of document signatures to infer and compute similarities between two Web pages. In the context of this paper, the term of "document signatures" is not equivalent to "signature files" (as in Witten, et al. [24]), which generally refers to strings of bits created by hashing all the terms in the document. In the present work, the signature approach relies on identifying page features and characteristics that not only identify the Web page, but also allow quantifying the change magnitude based on a comparison with a previously computed signature. As of now, the approach considers four Web page features:

Paragraph Checksums. This metric is used to determine content changes. By recording a checksum for each paragraph, this approach has a finer granularity than is possible with a page checksum. While they provide an idea about the degree of change to the page content, they also provide an idea of which pieces of text changed.

Headings. This metric is used to determine content and presentation changes. Headings typically highlight important text and titles. Changes to headings may indicate changes to the focus or perspective of a page. However, they may also reflect on how the document is divided and the information grouped and presented to the reader. Thus they provide clues about presentation changes.

Links. This metric is used to determine structural changes of the Web page. Since the value of hypertext documents depends not only on the document's contents, but also on the navigation provided by its links, it is important to analyze this connectivity. There are two components to links. On one hand there is an invisible component, namely the documents accessible through the links. On the other hand the visual text or image of the link anchors provides information about the contents of the destination pages. While the page might appear visually the same, the links might have changed to point to different places rendering the page inappropriate.

Keywords. We use reader-provided keywords in order to determine content changes of the Web page. In the current version, keyword presence is used as the feature to be identified. In future versions the effectiveness of more complex techniques such as TFIDF (Term Frequency Inverse Document Frequency) could be explored to determine the degree of change. Additionally, we are considering (but have not yet implemented) evaluating different algorithms for automatic keyword generation. Keywords provided by users are good at distinguishing relevant pages from non-relevant pages [16]. The question remains as to whether users will provide them.

The approach also records a *global checksum* for the whole page in order to diminish the possibility of false negatives. There are some changes to Web pages that do not affect any of the previous metrics. For example, a change to an image source would not be reflected in any of the current metrics as it is not part of the text, headings, links or keywords of the document. In this case the global checksum provides a last resort to point out changes.

Combining the results of the various document signatures into an single metric of change is difficult. As previously mentioned, some Web page changes are easily perceived while others are not. Some changes might alter the visual appearance of a page while maintaining the same structure and content. Other changes might change the links while maintaining exactly the same appearance. As already acknowledged, the relevance of change is situation dependent and no single metric will match all situations. We will return to the particular algorithms explored to compute this overall notion of change after a description of the system and interface being developed.

4. THE SYSTEM

The Walden's Paths Path Manager is a system implemented in Java capable of checking a list of Web pages for relevant changes. It takes a path file as input and checks all the pages specified in the path. Alternatively, an HTML file can be specified as input, whether it is located locally (like a bookmark list) or remotely (specified by a URL). The Path Manager interprets links in the HTML file as a list of URLs to check. In order to detect and assess the possible changes, the system retrieves all the pages from the Web, then parses their contents and creates a new signature for each page. The page signatures obtained are compared with the previously computed signatures. Finally, based on the comparison results, the system presents users with an assessment of the relevance of the overall change of each Web page. The user (the path's administrator) can then review each page individually and if there are no relevant changes, the user can validate the current state of the pages.

In order to compute the magnitude of the change, or the distance between two documents, a comparison is made between the signatures of the current version and those previously stored. Currently the Path Manager records three versions of the signature—the original, the last valid, and the latest time the path was checked. The original signature is the first signature obtained. It is used to give a sense of the total amount of change since the page was selected. The last valid signature corresponds to the last time that the user reviewed the changes and validated the pages. As time passes the user might update the valid signature as the pages change. This signature represents functional state of the page. Finally, the latest signature corresponds to the last time that the Path Manager retrieved and analyzed the pages, regardless if the user validated their state or not. Using these three signatures the Path Manager can indicate the amount of recent and long-term change for a page.

4.1 Scenario of Use

Figure 1 shows the initial state of the interface just after selecting a path to check.

The Path Manger extracts the pages from the path file and presents them to the user. Pages are identified by either their title or, when the page title is not available, their URL. Colors are used to represent the magnitude of the change. In this case pages are shown in blue, meaning that the current state of the page is unknown and needs to be checked. The flag at the left is used to indicate if any change has occurred. Since at this point that is still unknown, the flag is shown as hanging.

At this point the user can specify what algorithm and signature to use in order to check the pages in the list. There are two algorithms implemented, Johnson's algorithm and the proportional algorithm. As for the signatures, the Path Manager maintains three signatures: the original signature, the valid signature, and the latest signature.

Figure 2 shows the Path Manager relevance assessment of the changes in the pages using the proportional algorithm and comparing the signature with the last valid signature.

For each page a red flag is shown fluttering whenever the global checksum has changed. If the global checksum is the same, then a

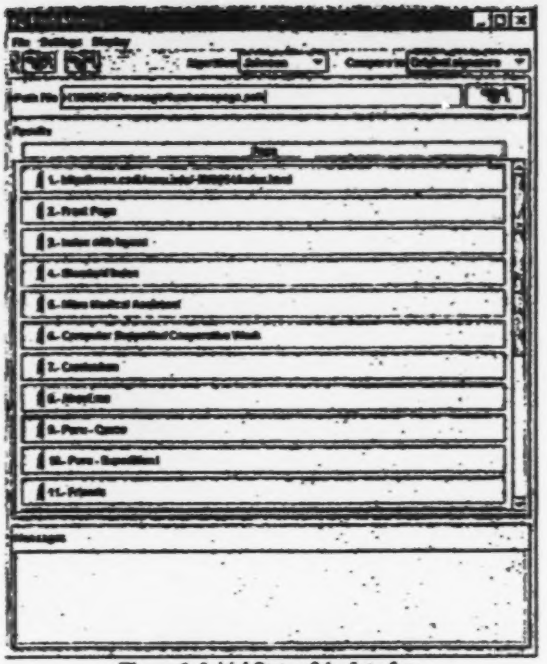

Figure 1. Initial State of the Interface.

blue hanging flag is shown next to the page title. The flags work as a boolean detection of even the slightest changes. In turn, the degree of relevance of the change is encoded by the color of the page title or URL. In particular, black means that there is no relevant change based on the algorithm chosen. Green, yellow, and red text indicate low, medium, and high degrees of changes considered relevant by the algorithm. The coloring scheme is based on user-defined trigger levels, which in turn could be used for other purposes such as not showing in the path a page that has changed more than the medium level.

At this point the user might choose to validate the state of the pages by clicking on the red "Valid" button. Alternatively the user might wonder about what kinds of changes prompt the system to assess the overall relevance of the changes. In order to support the more inquisitive user, the Path Manager can display the amount of change to the particular document signatures used in the relevance assessment. Figure 3 shows the view of the different metrics used in the relevance assessment.

The specific change metrics are presented to the right of the page identification. In addition, the user can get more information on each Web page by selecting the page identification. Figure 4 shows the detailed view of the change metrics for a particular page. In this case the bottom page from Figure 3 was chosen (page 11 in the figure). This view provides the assessment of change to the page at the bottom and information about the use of the page above. For the user to be able to assess the relevance of the change, they may need to see both the content of the page and the context of its use. The top of the detailed view for a page presents properties of the page within the path, such as the cache strategy, annotations and visibility of the page.

BEST COPY AVAILABLE

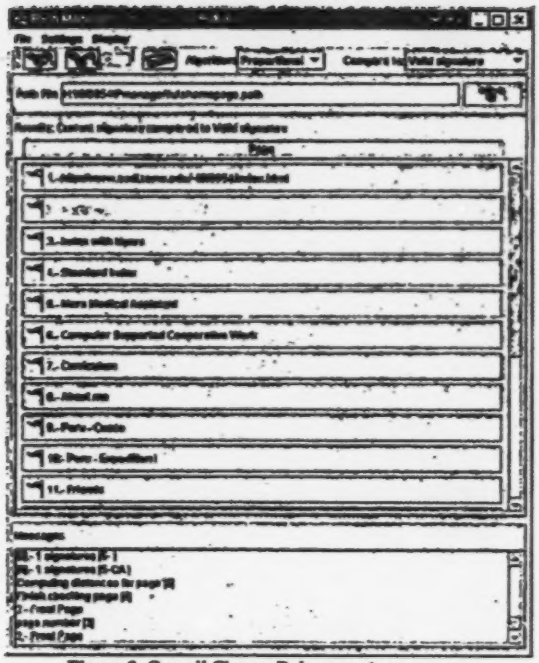

Figure 2. Overall Change Relevance Assessments.

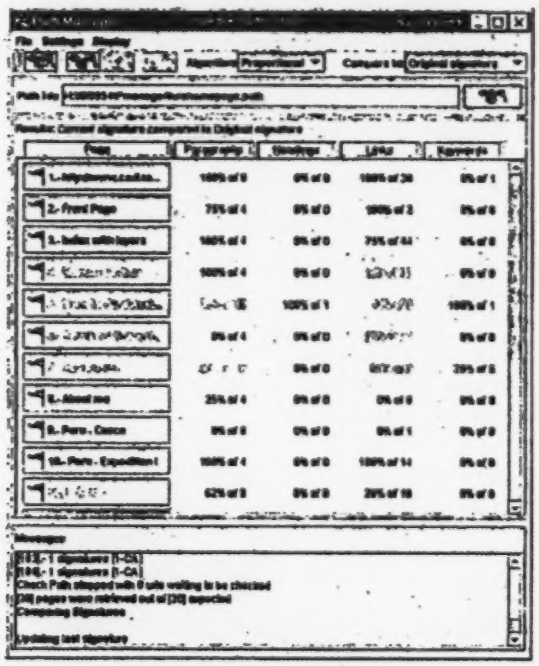

Figure 3. View of Change Metrics.

The system will also display the page in a Web browser at the user's request. At this point the user, who might not be the same person who authored the path, might judge the page inappropriate and choose to hide the page from the viewers. While this action prevents the material from being shown to readers of the path, it does not delete the page from the path, leaving that decision to the path author. In case there have been connection or retrieval problems, the manager can prompt the system to check this page again.

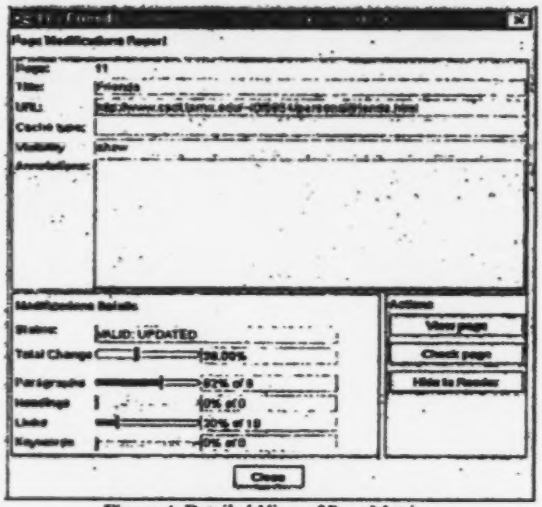

Figure 4. Detailed View of Page Metrics.

In addition the user asks for metadata about the path as a whole by selecting the information button, located to the right of the path file in the main interface (Figures 1-3). Figure 5 shows the path metadata presented to the user.

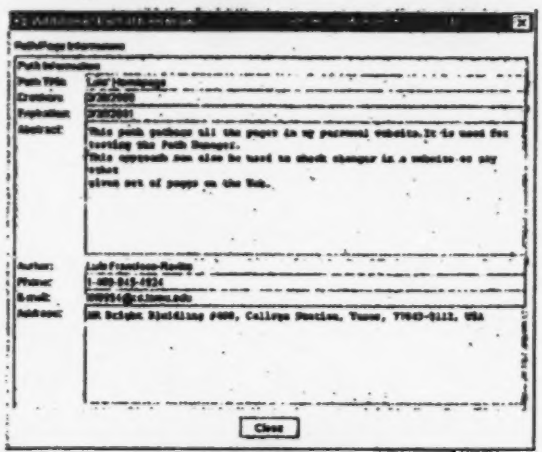

Figure 5. Path information.

Given this information, the user of the Path Manager can contact the path author in order to inform him/her of the changes or to perform and coordinate corrective measures in order to re-establish the desired function of the path.

BEST COPY AVAILABLE

4.2 Web Page Retrieval and Connectivity

An issue for the Path Manager is the connectivity required to retrieve a set of Web pages. Connection times are never constant, as they depend on many variables out of the system or users' control. Some pages are returned rather quickly, while others take longer to return. Also varying connection times means that, sometimes when a Web page seems to take too long to load, canceling the retrieval and immediately restarting the process results in the Web page being loaded faster.

Were the Web pages in a path to be checked sequentially, retrieval problems of a single page would block the retrieval of all consequent pages. Therefore the Path Manager architecture has been designed as a multi-threaded process in order to avoid possible blocks and expedite retrieval. In this scheme, each page is retrieved and analyzed in an independent thread. The user controls the maximum number of simultaneous threads.

Even using independent threads, there are many possible problems when retrieving pages from the Web such as no response, slow connections, and very long pages. In order to deal with these situations the system recognizes three states within each individual retrieval thread:

1. *Connection state*: the system is attempting to contact the Web server hosting the specified Web page.
2. *Retrieval state*: during this state the Web page has been located and its contents are being downloaded.
3. *Analysis state*: all contents have been retrieved and now the system is parsing and analyzing the contents.

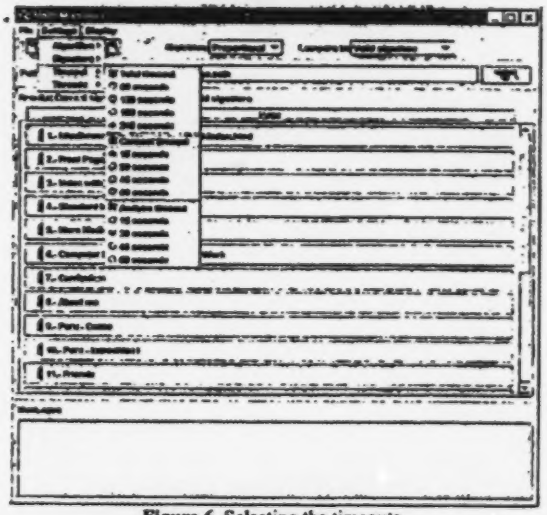

Figure 6. Selecting the timeouts.

The user can set different timeouts for the connection and retrieval states, which every thread must finish before they expire. The whole path checking process can also be interrupted by a general timeout or by the user clicking the stop button. Figure 6 shows the selection of the timeouts.

Each thread evaluates one page at a time. Once it either successfully or unsuccessfully evaluates the change to that page, the thread will pick another from the set to be checked until all pages have been successfully checked or a general timeout has occurred.

In any case that the Path Manager cannot assess the relevance of the changes, the page titles are shown in blue. Four different shades of blue are used to denote different reasons that the relevance assessment was not successful:

- The page is yet to be checked, i.e., no connection has been attempted.
- There was a network problem, typically a timeout during the connection or retrieval states.
- The general timeout expired before the analysis was completed.
- There were problems that could not be identified.

Because long paths encounter temporary connectivity problems that are resolved quickly, the user can also tell the Path Manager to check only those pages where the assessment of change was not successful.

4.3 Algorithms

We are investigating two different algorithms for categorizing and combining the changes found by the document signatures: one based on Johnson's work [10], and another we call the proportional algorithm.

4.3.1 Johnson's Algorithm

We first implemented a variation on Johnson's algorithm [10], [11] to compute the distance between two documents. As previously mentioned, Johnson only used paragraphs, headings and keywords. In order to take into account structural changes, in our implementation of his approach, we included links as an additional metric. For each signature, additions, deletions and modifications are identified. The distance metric is computed as follows.

$$D = PD + HD + LD + KD$$

Where:

D	– Distance
PD	– Paragraph Distance
HD	– Headings Distance
LD	– Links Distance
KD	– Keywords Distance

In turn

$$PD = \frac{PW * [(\#Pmod * PWmod) + (\#Padd * PWadd) + (\#Pdelete * PWdelete)]}{\#P}$$

Where:

#P	– Number of Paragraphs
#Pmod	– Number of Paragraph Modifications
#Padd	– Number of Paragraph Additions
#Pdelete	– Number of Paragraph Deletions
PW	– Paragraphs Weight
PWmod	– Paragraph Modifications Weight
PWadd	– Paragraph Additions Weight
PWdelete	– Paragraph Deletions Weight

Similarly:

$$HD = \frac{HW * [(\#Hmod * HWmod) + (\#Hadd * HWadd) + (\#Hdelete * HWdelete)]}{\#H}$$

$$LD = \frac{LW * [(\#Lmod * LWmod) + (\#Ladd * LWadd) + (\#Ldelete * LWdelete)]}{\#L}$$

BEST COPY AVAILABLE

The keyword computation is slightly simpler since it is only important to detect if a keyword is missing.

$$KD = \frac{KW * [\,Kdelete * KWdelete\,]}{\#K}$$

Johnson developed his algorithm to support Web-based tutorials. In his application, the results of the algorithm were used by the system in deciding whether a page should be displayed to the student—i.e., whether it continued to resemble the page selected by the tutorial's author. Johnson's algorithm includes the ability to weight modifications, additions, and deletions independently. However, since it is supporting a computer system and not a human user, it does not provide normalized results; this makes the results more difficult for a user to interpret. In particular, this makes the selection of cutoff values and the visualization of change somewhat unintuitive.

4.3.2 Proportional Algorithm

This signature-based approach provides a simpler computation of the distance. The change to each individual signature is computed as follows:

PD = (#Pchanges / #P) * 100
#Pchanges = #Pmods + #Padds + #Pdeletes

HD = (#Hchanges / #H) * 100
#Hchanges = #Hmods + #Hadds + #Hdeletes

LD = (#Lchanges / #L) * 100
#Lchanges = #Lmods + #Ladds + #Ldeletes

KD = (#Kchanges / #K) * 100
#Kchanges = #Kmods + #Kadds + #Kdeletes

The overall page distance is:

D = (#Tchanges / #T) * 100

#T = #P + #H + #L + #K
#Tchanges = #Pchanges + #Hchanges + #Lchanges + #Kchanges

Throughout the proportional algorithm, the number of paragraphs, headings, links, or keywords is taken to be the maximum of the two signatures being compared. Thus, whether a page goes from one to four paragraphs or from four paragraphs to one, #P = 4.

The proportional algorithm provides a normalized and symmetric distance that is easier to use for different sets of Web pages. The normalized and symmetric properties of the distance measurement facilitate providing the user with a visualization or listing of the different changes. This, in turn, allows the user to more effectively evaluate the page changes without having to actually review all the pages in the path.

5. THE PERCEPTION OF CHANGE

In order to inform and evaluate the approach and design of systems that aid in the automatic assessment of change relevance, we conducted a study to observe how humans perceive changes of Web pages. (We give a brief overview of the study here. For details consult the companion paper [5].)

This study covered three kinds of change: content, structure and presentation. In particular, content changes included modifications to the text presented, structure changes referred to modifications of URLs and the anchor text for links, and presentation changes included modifications to colors, fonts, backgrounds, spatial positioning of information, or combinations of these.

Web pages were selected from paths previously created by teachers and from personal bookmarks. Each page was modified to reflect a single type of change, and was classified by the magnitude and the type of change.

5.1 Methodology

The experiment consisted of presenting the participant with two versions of a Web page, the original version and a possibly modified version (some were unmodified copies of the original). The person was then asked to evaluate the magnitude of change.

The goal of the study was to address the following three questions:

1. Do people view the same changes in a different way when given different amounts of time to analyze the pages?
2. What kinds of changes are easily perceived?
3. Of what kind of changes do users want to be notified?

Before the evaluation we familiarized the participants with the testing software and the kinds of change. Additionally we provided a context for evaluating the changes in the Web pages by providing a scenario. The participant was asked to act the role of the Information Facilitator in a K-12 school (Kindergarten to High School). Teachers have chosen pages from the Web to teach their classes and it is the participant's responsibility to check for changes.

The first task in the study was to fill in a pre-evaluation questionnaire collecting demographic data about the participants and their computer literacy. The study was then divided into three phases:

1. In *phase I*, the person was given 60 seconds to view each pair of Web pages. In this phase the system presented the participant 8 cases, which included examples from all of the different kinds of change. The system also identified the changes to the participant. The objective of this phase was to provide training, and the answers to the question allowed us to address the third question.
2. In *phase II*, the person was only allowed to view the page pairs for 15 seconds before evaluating them. There were 31 pairs in this phase. This phase addressed the first and second questions.
3. In *phase III*, the person was given 60 seconds to view the Web pages. There were 31 pages in this phase, which also addressed the first and second questions.

In each phase, once the person viewed a Web page pair, s/he was presented with five questions:

1. From the Content perspective, the degree of change is:
2. From the Structure perspective, the degree of change is:
3. From the Presentation perspective, the degree of change is:
4. Overall, how significant are the changes?
5. If this page were in my bookmark list, I would like to be notified when changes like these occur.

Questions 1-4 were answered in a 7-stop scale ranging from "none" to "moderate" to "drastic". Question 5 was answered in a 7-stop scale ranging from strongly disagree to strongly agree.

The final task in the study was to fill a post-evaluation questionnaire that gathered the participant's general comments about how easy it was to assess the magnitude of changes in Web pages, and about the evaluation software.

In all phases the testing software controlled the presentation of the Web pages directly. The time required to record the participant answer was not controlled. The total time for the study varied from 1 to 2 hours per person, depending on the time the participants took in filling in the answers, and whether they decided to rest between phases.

5.2 Target Population

In order to conduct the study, we recruited adults, specifically students at Texas A&M University. Subjects were divided into two groups. Pages in Phase II for one group were used as the pages in Phase III for the other group. This allows comparing the results for the same page when given 15 or 60 seconds for observation.

5.3 Study Results

The results of Phase I, when the subjects were provided textual descriptions of what had changed, give the clearest indication of what people consider change and of what they would like to be notified. For content changes, as the percent of paragraphs changed, the perception of overall change also increased, as did the subjects' desire to be notified of the change. Interestingly, as the degree of structural changes increased, the perception of the overall change did not increase but the desire to be notified of the change did. For similar percentages of content and structure change, subjects rated the content changes higher in overall change but lower in desire to be notified.

In Phases II and III, time did not alter the perception of content change but was seen to play a large role in the identification of structural change. This is likely due to the visibility of content changes and the invisibility of structure changes—subjects commented in the exit survey about how difficult and time-consuming it was to detect structure change. As with the results of Phase I, subjects desire to be notified of perceived changes in structure were higher than their desire to be notified for similarly perceived changes in content.

In Phases II and III, subjects rated low and medium changes in content similarly, but rated those pages that had drastically changed quite a bit higher for all questions. Structural changes saw a similar but less extreme jump in ratings when almost all the links had been changed.

Presentation changes were not considered by amount of change but by type. Subjects did not seem to notice changes to fonts, while changes to background color and navigational images were noticed but rated low as contributing to overall change or desire to be notified. Changes in the arrangement of material on a page was noticed by subjects but interpreted differently depending on the amount of time they had to look at the page. With only 15 seconds, the rearrangement was considered a content change and there was a large desire to be notified while with 60 seconds the subjects recognized that the material was the same but moved and they had a lower desire to be notified. Finally, when many presentation features were changed at once, subjects rated the change as high on all metrics and wanted to be notified.

5.4 Implications for the Path Manager

The results of the study indicate that including links in metrics for overall change will better match our subject's evaluation of change and desire to be notified. The results also show that presentation changes were viewed as largely unrelated to the function of the page, except in extreme cases. Finally, a more detailed analysis of the results may provide weightings for combining the results of the four signatures into an overall view of change.

6. CHALLENGES AND LIMITATIONS

An issue with using document signatures based on HTML tags is that not every page creator or Web page authoring tool uses the HTML tags consistently. In the case of headings, page creators often choose to modify the visual appearance of the text by using tags such as FONT or resorting to images. This imposes the requirement of implementing smarter parsers. We are currently attempting to address some of these by augmenting the system to infer what is conceptually a heading and where paragraphs begin and end, and to identify different types of changes to links.

A limitation of the current Path Manager is that no indirection is supported. When faced with Web pages containing frames, the Path Manager does not check the pages contained in the frames. The same is true for other tags such as images or links. There is no retrieval of the pages specified in the SRC or HREF fields unless they are also included explicitly in the path.

Another limitation of the Path Manager is that it does not monitor any JavaScript or other page behaviors. This is in part due to the complexity of the required parser and to the fact that this remains a moving target, where specifications and support vary constantly over time and browser type.

Finally, while the Path Manager can parse dynamically generated pages returned by CGIs, it does not recognize that they are dynamic and therefore variable by nature. Augmenting the system to recognize these, could enable separate treatment for such pages.

7. CONCLUSIONS

Maintenance of distributed collections of documents remains a challenging and time-consuming task. People must monitor the documents for change and then interpret changes to the documents in the context of the collection's goals.

The Walden's Paths Path Manager supports the maintenance of collections of Web pages by recognizing, evaluating, and informing the user of changes. The evaluation of change is based on document signatures of the paragraphs, headings, links, and keywords. The Path Manager keeps track of original, last valid, and last collected signatures so users can determine both long-term and short-term change depending on their particular concern.

The Path Manager has been designed to work in a distributed environment where connectivity to documents is unpredictable. Its architecture and instantiation provide users with control over system resources consumed. Also, the system provides feedback about documents that are not successfully evaluated.

Particular to uses for Walden's Paths, the Path Manager provides access to information about the use of the documents in a path and to metadata about the path that may be important to determining the relevance of particular changes. Users may also mark pages so

that they remain in the path but the Walden's Paths server will not display them to readers of the path.

A study of the perception of changes to Web pages indicated the desire for structural changes to be included in the determination of overall change. The study also showed that presentation changes were largely considered irrelevant. Current work on the Path Manager aims to overcome difficulties with the inconsistencies and indirection found in Web documents.

8. ACKNOWLEDGEMENTS

This material is based upon work supported by the National Science Foundation under Grant Numbers IIS-9812040 and DUE-0085798.

9. REFERENCES

[1] Armstrong, R., Freitag, D.M, Jachims, T., & Mitchell T. WebWatcher: A Learning Apprentice for the World Wide Web, in Working Notes of AAAI Spring Symposium on Information Gathering from Heterogeneous Distributed Environments (Stanford University CA, March 1995) AAAI Press, 6-12.

[2] Bush, V. As We May Think. The Atlantic Monthly, (August 1945), 101-108.

[3] Brewington, B. & Cybenko, G. How Dynamic is the Web, in Proc. of WWW9—9th International World Wide Web Conference, IW3C2, (2000) 264-296.

[4] Douglis, F., Ball, T., Chen, Y., and Koutsofios, E. The AT&T Internet Difference Engine: Tracking and Viewing Changes on the Web, in World Wide Web, 1(1) 27-44 (January 1998).

[5] Francisco-Revilla, L., Shipman, F.M., Karadkar, U., Furuta, R., and Arora, A. Changes to Web Pages: Perception and Evaluation. To appear in Proc. Hypertext 2001, (Åarhus, Denmark, August 2001).

[6] Furuta, R., Shipman, F., Francisco-Revilla, L., Karadkar, U., & Hu, S. Ephemeral Paths on the WWW: The Walden's Paths Lightweight Path Mechanism. WebNet (1999), 409-414.

[7] Furuta, R., Shipman, F., Marshall, C., Brenner, .D., & Hsieh, H. Hypertext Paths and the World-Wide Web: Experiences with Walden's Paths, in Proc. of Hypertext'97 (Southampton U.K., April 1997). ACM Press, 167-176.

[8] Giles, L., Bollacker, K., & Lawrence, L. CiteSeer: An Automatic Citation Indexing System, in Proc. of DL'98 (Pittsburgh PA, June 1998). ACM Press, 89-98.

[9] Joachims, T., Mitchell, T., Freitag, D., & Armstrong, R. WebWatcher: Machine Learning and Hypertext, Fachgruppenterffen Maschinelles Lernen. Dortmund, Germany, August 1995.

[10] Johnson, D.B. Enabling the Reuse of World Wide Web Documents in Tutorials. PhD. Dissertation, Dept. of computer Science and Engineering. University of Washington, Seattle, WA. 1997

[11] Johnson, D.B., & Tanimoto, S.L. Reusing Web Documents in Tutorials with the Current-Documents Assumption: Automatic Validation of Updates, in Proc. of EDMEDIA'99 (Seattle WA, June 1999). AACE, 74-79.

[12] Karadkar, U., Francisco-Revilla, L., Furuta, R., Hsieh, H., & Shipman, F. Evolution of the Walden's Paths Authoring Tools, in Proceedings of WebNet 2000--World Conference on the WWW and Internet (San Antonio, TX, October 30--November 4, 2000) AACE, 299-304.

[13] Levy, D.M. Fixed or Fluid? Document Stability and new Media, in Proc. of the European Conference on Hypertext Technology '94 (Edinburgh Scotland, September 1994). ACM Press, 24-41.

[14] Lieberman, H. Letizia: An Agent That Assists Web Browsing. International Joint Conference on Artificial Intelligence (Montreal Canada, August 1995). Morgan Kaufman, 924-929.

[15] Lieberman, H. Autonomous Interface Agents, in Proc. of CHI'97 (Atlanta GA, March 1997). ACM Press, 67-74.

[16] Pazzani, M., & Billsus, D. Learning and Revising Reader Profiles: The Identification of Interesting Web Sites. Machine Learning 27 (1997), Kluwer Academic Publishers, 313-331.

[17] Pazzani, M., Muramatsu, J., & Billsus, D. Syskill and Webert: Identifying interesting Web sites, in Proc. of AAAI'96 (Portland Oregon, August 1996). American Association for Artificial Intelligence, 54-59

[18] Shipman, F., Furuta, R., Brenner, .D., Chung, C., & Hsieh, H. Using Paths in the Classroom: Experiences and Adaptations, in Proc. Hypertext'98 (Pittsburgh PA, June 1998). ACM Press, 167-176.

[19] Shipman, F., Marshall, C., Furuta, R., Brenner, .D., Hsieh, H., & Kumar, V. Creating Educational Guided Paths over the World-Wide Web, in Proc. of ED-TELECOM'96 (Boston MA, June 1996), AACE, 326-331.

[20] Starr, B., Ackerman, M.S., & Pazzani, M. Do-I-Care: a collaborative Web agent, in Proc. of CHI'96, (Vancouver Canada, April 1996), ACM Press, 273-274.

[21] Starr, B., Ackerman, M.S., & Pazzani, M. Do-I-Care: Tell Me What's Changed on the Web, in Proc. of the AAAI Spring Symposium on Machine Learning in Information Access (Stanford CA, March 1996).

[22] URL-Minder. Available at http://www.netmind.com/html/url-minder.html

[23] WatzNew. Available at http://www.watznew.com

[24] Witten, I.H., Moffat, A., and Bell, T.C., Managing Gygabytes. Compressing and Indexing Documents and Images, 2nd Edition, Morgan Kaufman, San Francisco, CA, 1999.